PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04L 12/56

A2

(11) International Publication Number: WO 99/13620

(43) International Publication Date: 18 March 1999 (18.03.99)

(21) International Application Number: PCT/SE98/01585

(22) International Filing Date: 7 September 1998 (07.09.98)

(30) Priority Data:
9703293-2 9 September 1997 (09.09.97) SE

(71) Applicant (*for all designated States except US*): SICS [SE/SE]; Swedish Institute of Computer Science, Box 1263, S-164 29 Kista (SE).

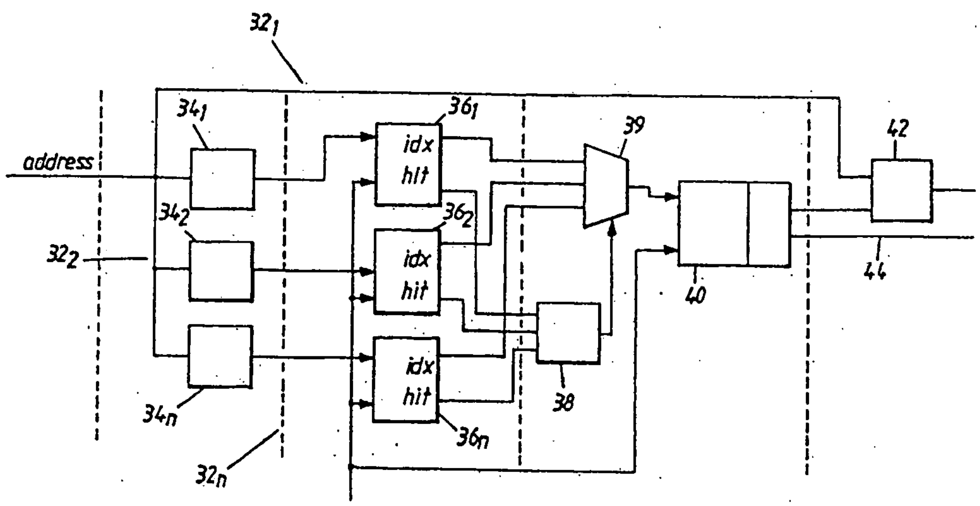
(72) Inventors; and
(75) Inventors/Applicants (*for US only*): SJÖDIN, Peter [SE/SE]; Nyodlingsvägen 14B, S-191 40 Sollentuna (SE). MOESTEDT, Andreas [SE/SE]; Fatburskvarnsgata 2, S-118 64 Stockholm (SE).

(74) Agent: ASKERBERG, Fredrik; L.A. Groth & Co. KB, P.O. Box 6107, S-102 32 Stockholm (SE).

(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, HR, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

**Published**
*Without international search report and to be republished upon receipt of that report.*

(54) Title: A LOOKUP DEVICE AND A METHOD FOR CLASSIFICATION AND FORWARDING OF PACKETS IN PACKET-SWITCHED NETWORKS




(57) Abstract

The present invention relates to a lookup device and a method for classification and forwarding of packets in packet-switched networks, wherein each packet comprises a packet header comprising a number of fields, wherein several fields in the packet header together form a packet identifier. The lookup device (30) comprises n parallel hashing units ($32_1$, $32_2$, ... $32_n$), wherein n is an integer and n≥2, for computing, for each packet, a first index by hashing the packet identifier, and in dependence of the first index either directly or indirectly obtaining a packet identifier and forwarding information for the destination for said packet from one of at least n memories . The n hashing units ($32_1$, $32_2$, ... $32_n$) are processing the same packet identifier at a time. The lookup device (30) also comprises a comparator (42) connected either directly or indirectly to at least one of said memories and to an input to said n hashing units ($32_1$, $32_2$, ... $32_n$) for comparing the packet identifier input to the n hashing ($32_1$, $32_2$, ... $32_n$) and the packet identifier output from said memory. When the compared packet identifiers match, the forwarding information for the packet is obtained from said memory.

*FOR THE PURPOSES OF INFORMATION ONLY*

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| AL | Albania | ES | Spain | LS | Lesotho | SI | Slovenia |
| AM | Armenia | FI | Finland | LT | Lithuania | SK | Slovakia |
| AT | Austria | FR | France | LU | Luxembourg | SN | Senegal |
| AU | Australia | GA | Gabon | LV | Latvia | SZ | Swaziland |
| AZ | Azerbaijan | GB | United Kingdom | MC | Monaco | TD | Chad |
| BA | Bosnia and Herzegovina | GE | Georgia | MD | Republic of Moldova | TG | Togo |
| BB | Barbados | GH | Ghana | MG | Madagascar | TJ | Tajikistan |
| BE | Belgium | GN | Guinea | MK | The former Yugoslav Republic of Macedonia | TM | Turkmenistan |
| BF | Burkina Faso | GR | Greece | | | TR | Turkey |
| BG | Bulgaria | HU | Hungary | ML | Mali | TT | Trinidad and Tobago |
| BJ | Benin | IE | Ireland | MN | Mongolia | UA | Ukraine |
| BR | Brazil | IL | Israel | MR | Mauritania | UG | Uganda |
| BY | Belarus | IS | Iceland | MW | Malawi | US | United States of America |
| CA | Canada | IT | Italy | MX | Mexico | UZ | Uzbekistan |
| CF | Central African Republic | JP | Japan | NE | Niger | VN | Viet Nam |
| CG | Congo | KE | Kenya | NL | Netherlands | YU | Yugoslavia |
| CH | Switzerland | KG | Kyrgyzstan | NO | Norway | ZW | Zimbabwe |
| CI | Côte d'Ivoire | KP | Democratic People's Republic of Korea | NZ | New Zealand | | |
| CM | Cameroon | | | PL | Poland | | |
| CN | China | KR | Republic of Korea | PT | Portugal | | |
| CU | Cuba | KZ | Kazakstan | RO | Romania | | |
| CZ | Czech Republic | LC | Saint Lucia | RU | Russian Federation | | |
| DE | Germany | LI | Liechtenstein | SD | Sudan | | |
| DK | Denmark | LK | Sri Lanka | SE | Sweden | | |
| EE | Estonia | LR | Liberia | SG | Singapore | | |

# A LOOKUP DEVICE AND A METHOD FOR CLASSIFICATION AND FORWARDING OF PACKETS IN PACKET-SWITCHED NETWORKS

## Technical field of the invention

The present invention relates to a lookup device for classification and forwarding of packets in packet-switched networks. The present invention also relates to a method for classification and forwarding of packets in packet-switched networks.

## Description of related art

The growth of the Internet has led to a situation where bandwidth is becoming a scarce resource. One reason for this is that the IP routers - the packet switches in the Internet - are not powerful enough to handle the traffic that aggregates at the switching points. The current trend for dealing with this problem is to relieve routers from some of the burden of switching traffic, and instead use switches of different kinds, such as FDDI switches, ATM switches, and Ethernet switches. This turns out to be a more cost effective solution, since the price for switching capacity is much lower than the price for routing capacity.

One of the main limiting factors for performance in an IP router, compared to a switch is often claimed to be the processing of incoming packets. When an IP packet arrives at an input port of a router, the packet needs to be examined and classified, and based on the classification the packet is forwarded to an output port. The packet classification operation consists of analysing information in the packet header (at least the destination address needs to be examined), and performing a lookup operation to obtain the information required to forward the packet to its next hop. In principle, the same kind of classification needs to be performed by a switch, but the operation is generally thought to be more complicated for an IP packet than for an ATM cell or an Ethernet frame. A common lookup method is to use a hashing scheme.

In the article "A Comparison of Hashing Schemes for Address Lookup in Computer Networks", by R. Jain, IEEE Transactions on Communication, vol. 40, No. 10, pp. 1570--

1573, 1992, is disclosed different hashing methods. The described hashing methods are:

1) hashing using address bits,
2) hashing using CRC polynomials,
3) hashing using Fletcher checksum,
4) hashing using another checksum, and
5) hashing using XOR folding.

The article "Large-scale and High-speed Interconnection of Multiple FDDIs using ATM-based Backbone LAN", by T. Tsukakoshi, O. Takada, T. Murakami, M. Terada, M. Yamaga, IEEE INFOCOM '92, vol. 3, pp. 2290-2298, May 1992, describes a solution to the problem that a hash function can map several identifiers into the same table location. The hashing mechanism according to this solution puts all entries in the same memory, and calculates the hash value a variable number of times until no collision occurs.

Typically hash tables are implemented as a table of lists, where each table entry is a list of identifiers that share that index. The disadvantage with such an organisation is that it requires repetitive accesses to memory in order to do a lookup, lowering performance. Furthermore, many such schemes rely on only one hash function, so rehashing has to be performed if the distribution gets too skewed.

Summary of the invention

The object of the present invention is to solve the above mentioned problems and to provide a lookup device for classification and forwarding of packets, wherein each packet comprises a packet header comprising a number of fields, wherein several fields in the packet header together form a packet identifier. This object is achieved by providing the lookup device defined in the introductory part of Claim 1 with the advantageous features of the characterizing part of said Claim.

The lookup device according to the present invention comprises n parallel hashing units, wherein n is

an integer and $n \geq 2$, for computing for each packet, a first index by hashing the packet identifier, and in dependence of the first index either directly or indirectly obtaining a packet identifier and forwarding information for the destination for said packet from one of at least n memories, wherein the n hashing units are processing the same packet identifier at a time. The lookup device also comprises a comparator connected either directly or indirectly to at least one of said memories and to an input to said n hashing units for comparing the packet identifier input to the n hashing units and the packet identifier output from said memory. When the compared packet identifiers match the forwarding information for the packet is obtained from said memory. The main advantage with this design is that a new packet identifier can be looked up in each memory cycle time. Another advantage with this design is that it allows several table lookups to be performed in one memory cycle time, since the lookups are performed in parallel.

Advantageously, each hashing unit comprises a hash function means for computing said first index, and a hash memory connected to said hash function means.

Preferably, the lookup device makes use of n different hash functions, one hash function for each hash function means. Hereby is achieved that the need of rehashing is effectively decreased, and hopefully eliminated since the identifiers are spread by several independent hash functions.

Preferably, the n hashing units are ordered by priority, wherein the first hashing unit has the highest priority, and the n:th hashing unit has the lowest priority.

Advantageously, the first hash memory, representing the highest level in the lookup device, has the largest memory size, and the n:th hash memory, representing the lowest level in the lookup device, has the smallest memory size.

Preferably, the memory sizes for the n hash memories are decreasing substantially lineary. Hereby is achieved the most efficient memory usage.

Advantageously, all the memories are Static Random Access Memories (SRAM's) and/or Dynamic Random Access Memories (DRAM's).

Preferably, said first index function as an input to said hash memory giving a packet identifier and forwarding information for the destination and a hit flag as outputs. The lookup device also comprises a selecting means connected to the hit flag outputs of all n hash memories, a multiplexer connected to the packet identifier and forwarding information outputs of all n hash memories, wherein said comparator is connected to said multiplexer. A set hit flag indicates that there was an entry in the hash memory with the first index obtained by hashing the packet identifier, and the packet identifier from the hash memory with the highest priority with the hit flag set, if any, is used as input to said comparator via said multiplexer, whereby said comparator compares the packet identifier input to said hash function means and the packet identifier output from said multiplexer, and when the compared packet identifiers match, the forwarding information for the packet is obtained from the hash memory with the highest priority with the hit flag set.

According to another embodiment of the present invention said first index function as an input to said hash memory giving a second index and a hit flag as outputs. The lookup device also comprises a selecting means connected to the hit flag outputs of all n hash memories, a multiplexer connected to the second index outputs of all n hash memories, an address memory, storing all packet identifiers together with the forwarding information for the destination, connected to said multiplexer, wherein said comparator is connected to said address memory. A set hit flag indicates that there was an entry in the hash memory with the first index obtained when hashing the packet identifier, and the second index

from the hash memory with the highest priority with the hit flag set, if any, is used as input to said address memory, giving a packet identifier and the forwarding information as outputs. The comparator compares the packet identifier input to said hash function means and the packet identifier output from said address memory, and when the compared packet identifiers match, the forwarding information for the packet is obtained from said address memory.

Another object of the invention is to provide a method for classification and forwarding of packets, wherein each packet comprises a packet header comprising a number of fields, wherein several fields in the packet header together forms a packet identifier. The method comprises the following steps:

- to compute, for each packet, a first index by hashing the input packet identifier in n different, parallel paths, wherein n is an integer and $n \geq 2$;
- and in dependence of the first index either directly or indirectly obtaining a packet identifier and forwarding information for the destination for said packet from one of at least n memories;
- to compare the input packet identifier and the packet identifier output from the memory; and
- if the compared packet identifiers match to make use of the forwarding information obtained from said memory.

The main advantage with this method is that a new packet identifier can be looked up in each memory cycle time. Another advantage with this method is that it allows several table lookups to be performed in one memory cycle time, since the lookups are performed in parallel.

Advantageously, the computing step comprises the steps:

- to compute the first index by using different hash functions, one hash function for each path; and

- to use the first index as an input to a table, one of n different tables. Hereby is achieved that the need of rehashing is effectively decreased, and hopefully eliminated since the identifiers are spread by several independent hash functions.

Preferably, the n paths are ordered by priority, wherein the first path has the highest priority and the n:th path has the lowest priority.

Preferably, the first table, representing the highest level, has the largest size, and the n:th table, representing the lowest level, has the smallest size.

Advantageously, the sizes of the n tables are decreasing substantially lineary. Hereby is achieved the most efficient table usage.

Preferably, each table stores packet identifiers and forwarding information for the destination, and wherein each table outputs a hit flag, wherein a set hit flag indicates that there was an entry in the table with the first index obtained by hashing the packet identifier, and the packet identifier from the table with the highest priority with the hit flag set, if any, is used as input to said comparing step.

According to another embodiment of the method according to the present invention, said first index functions as an input to said table giving a second index and a hit flag as outputs. A set hit flag indicates that there was an entry in the table with the first index obtained when hashing the packet identifier, and the second index from the table with the highest priority with the hit flag set, if any, is used as input to an address memory giving a packet identifier as output, and said packet identifier is used as input to said comparing step.

Preferably, if a new packet identifier is to be added, it is initially hashed into the first path, and if a collision occurs, i.e. there is already a packet identifier with that first index in the first table, the two colliding packet identifiers are hashed into the

second path, and if a collision occurs in the i:th path, the colliding packet identifiers are hashed into the (i+1):th path, wherein $1 \leq i \leq n-1$. Hereby is achieved that only one comparison is needed for a full identifier lookup.

Advantageously, the method terminates for said packet identifier if the compared packet identifiers not match.

Preferably, the method terminates for said packet identifier if none of the n tables outputs a set hit flag.

Embodiments of the invention will now be described with a reference to the accompanying drawings, in which:

## Brief description of the Drawings

Figure 1 shows a schematic diagram of the fields in an IP packet header;

Figure 2 shows a schematic diagram of the hashing concept;

Figure 3 shows a block diagram of a lookup device according to the present invention; and

Figure 4 is a flow chart of the method according to the present invention.

## Detailed description of Embodiments

In figure 1 there is disclosed a schematic diagram of the fields in an IP packet header. The IP packet header comprises 12 different fields. As is disclosed in figure 1 these fields are: Version, IP Header Length, Type of Service, Total Length, Identification, Flags, Fragment Offset, Time to Live, Protocol, Header Checksum, Source Address, and Destination Address. It can also contain an Options field.

There are in principle two different types of IP packet classification: IP address lookup, which is used for forwarding of unicast packets based on their destination address, and identifier lookup, which is intended to be used for, for example, forwarding of multicast packets and flows of packets. IP multicast

addresses are not organized in a hierarchical structure. Identifier lookup is used when several fields in the packet header together form a packet identifier. Such an identifier has no hierarchical structure to it, and the identifier space is potentially very large. Therefore techniques such as hashing or CAM (Content Addressable Memory) are required for the lookup. The present invention is based on identifier lookup.

In the article "A Comparison of Hashing Schemes for Address Lookup in Computer Networks", by R. Jain, IEEE Transactions on Communication, vol. 40, No. 10, pp. 1570-1573, referred to above, is disclosed the basic theory underlying the hashing concept. Below and in reference to figure 2 will be given a small selection from this article.

In figure 2 there is disclosed a schematic diagram of the hashing concept. Basically, hashing allows us to chop up a big table into several small subtables so that we can quickly find the information once we have determined the subtable to search for. This determination is made by using a mathematical function, which maps the given key to hash cell i, as shown in figure 2. The cell i could then point us to the subtable of size ni. Given a trace of R frames with N distinct addresses and a hash table of M cells, the goal is to minimize the average number of lookups required per frame.

If we perform a regular binary search through all N addresses, we need to perform $1 + \log_2(N)$ or $\log_2(2N)$ lookup per frame. Given an address that hashes to i:th cell, we have to search through a subtable of ni entries, which requires only $\log_2(2n_i)$ lookups. The total number of lookups saved is S:

$$S = \sum_i r_i[\log_2(2N)-\log_2(2n_i)]$$

where $r_i$ is the number of frames that hash to the i:th cell, $\sum r_i = R$. The net saving per frame is F:

$$F = \sum_i -\frac{r_i}{R} \log_2\left(\frac{n_i}{N}\right) = \sum_i -q_i \log_2(P_i)$$

Here, $q_i = \frac{r_i}{R}$ denotes the fraction of frames that hash to i:th cell, and $p_i = \frac{n_i}{N}$ is the fraction of addresses that hash to i:th cell. The goal of a hashing function is to maximize the quantity $\sum -q_i \log_2(P_i)$.

In figure 3 there is disclosed a block diagram of a lookup device according to the present invention. The lookup device 30 is for classification and forwarding of packets in packet-switched networks, wherein each packet comprises a packet header (see figure 1) comprising a number of fields, wherein several fields in the packet header together forms a packet identifier. The lookup device 30 comprises n parallel hashing units $32_1$, $32_2$, ... $32_n$, wherein n is an integer and n≥2. Each hashing unit $32_1$, $32_2$, ... $32_n$ comprises a hash function means $34_1$, $34_2$, ... $34_n$, and a hash memory $36_1$, $36_2$, ... $36_n$ connected to said hash function means $34_1$, $34_2$, ... $34_n$. Each hash function means $34_1$, $34_2$, ... $34_n$ computes a first index by hashing the packet identifier. This first index is used as an input to said hash memory, giving a second index and a hit flag as outputs. A set hit flag indicates that there was an entry in a hash memory $36_1$, $36_2$, ... $36_n$ with the first index obtained when hashing the packet identifier. The lookup device 30 according to the present invention makes use of n different hash functions, one hash function for each hash function means $34_1$, $34_2$, ... $34_n$. This means that the lookup device 30 according to the present invention comprises several (n) parallel hash paths. All hashing units $32_1$, $32_2$, ... $32_n$ process the same packet identifier. Therefore a lookup for a given identifier will succeed in at most one of the paths, and therefore all paths can be searched in parallel. The lookup device 30 also comprises a selecting means 38 connected to the hit

flag outputs of all n hash memories $36_1$, $36_2$, ... $36_n$, and a multiplexer 39 connected to the second index outputs of all n hash memories $36_1$, $36_2$, ... $36_n$. The output from the selecting means 38 is connected to said multiplexer 39. The lookup device 30 also comprises an address memory 40, storing all the packet identifiers together with the forwarding information for the destination. Each second index input to said address memory 40 will give a packet identifier and the forwarding information for the destination as output. The lookup device 30 also comprises a comparator 42 connected to said address memory 40. The comparator 42 has also another input, supplied with the identifier input to all the n hash function means $34_1$, $34_2$, ... $34_n$. The comparator 42 compares the packet identifier input to the n hash function means $34_1$, $34_2$, ... $34_n$, and the packet identifier output from the address memory 40. If the compared packet identifiers match, the forwarding informaton for the packet is obtained from said address memory 40, via a line 44. If they do not match it was a false hit, indicating that the packet identifier was not present in the address memory 40. The hash calculation, the memory lookup, the table lookup and the comparison are all independent operations and can work in parallel, thus the lookup can easily be pipelined to increase the throughput.

Another embodiment of the lookup device according to the present invention does not comprise an address memory and does not make use of any second index. Instead the hash memories $36_1$, $36_2$, ... $36_n$ comprise the packet identifiers and the forwarding information. The packet identifier output from the hash memory with the highest priority with the hit flag set, if any, is used as input to said comparator. This embodiment is not disclosed in any figure. This embodiment comprises all the elements disclosed in figure 3, except the address memory 40.

The embodiment disclosed in figure 3 is preferred for large identifiers, because it saves memory to use a second level memory. What method is best depends on how

the design is used (i.e. size of identifiers, memory organization, etc.).

The advantages with these designs are twofold: first, it allows several table lookups to be performed in one memory cycle time, since the lookups are performed in parallel. Second, there are several hash functions, which effectively decrease, and hopefully eliminate, the need of rehashing since the identifiers are spread by several independent hash functions.

When an identifier is added to the lookup device 30 it is initially hashed into the first path. If a collision occurs, i.e. there is already an identifier with that index in the hash memory, the two colliding identifiers are hashed into the second path. The index in the first path where the collision occurred cannot be used for another identifier. If a collision occurs also in the second path the same procedure is repeated with the colliding identifiers moved to the third path, and so on.

There is a reason why a hash entry where a collision has occurred is not used any more. If a lookup hits in more than one path, then it is sufficient to only consider the hit in the path with the highest priority. So with this scheme, only one comparison with the real identifier has to be performed.

If a collision occurs in the last of paths, the lookup table will overflow. The larger the hash memories in the hash paths, the lower the probability that identifiers will collide.

The most efficient memory usage is obtained when the memory is divided into several hash paths. The hash paths should be organised hierarchically with the largest hash memory at the highest level and the smallest hash memory at the lowest level, preferably with the hash memory sizes for the n hash memories decreasing substantially lineary.

The lookup device 30 is preferably implemented using Static Random Access Memories (SRAMs) and/or Dynamic Random Access Memories (DRAMs) as memories.

The hash function means $34_1$, $34_2$, ... $34_n$ can be implemented using xor folding, which is probably preferred, being very simple and easy to vary.

In figure 4 there is disclosed a flow chart of the method according to the present invention. The method for classification and forwarding of packets, wherein each packet comprises a packet header (see figure 1) comprising a number of fields, wherein several fields in the packet header together form a packet identifier, begins at block 50. Thereafter, at block 52, the method continues to compute, for each packet, a first index by hashing the input packet identifier in n different, parallel paths, wherein n is an integer and $n \geq 2$. Thereafter, at block 54, the method continues by, in dependence of the first index, either directly or indirectly obtaining a packet identifier and forwarding information for the destination for said packet from one of at least n memories. Then, at block 56, the method continues by comparing the input packet identifier and the packet identifier output from the memory. Then, at block 58, the method continues, if the compared packet identifiers match, by making use of the forwarding information obtained from said memory. Then, at block 60, the method is completed.

The method according to the present invention can e.g. be implemented with a lookup device of the type disclosed in figure 3.

The invention is not limited to the embodiment described in the foregoing. It will be obvious that many different modifications are possible within the scope of the following Claims.

Claims

1. A lookup device (30) for classification and forwarding of packets, wherein each packet comprises a packet header comprising a number of fields, wherein several fields in the packet header together form a packet identifier, **characterized in** that the lookup device (30) comprises n parallel hashing units ($32_1$, $32_2$, ... $32_n$), wherein n is an integer and n≥2, for computing, for each packet, a first index by hashing the packet identifier, and in dependence of the first index either directly or indirectly obtaining a packet identifier and forwarding information for the destination for said packet from one of at least n memories, wherein the n hashing units ($32_1$, $32_2$, ... $32_n$) are processing the same packet identifier at a time, and a comparator (42) connected either directly or indirectly to at least one of said memories and to an input to said n hashing units ($32_1$, $32_2$, ... $32_n$) for comparing the packet identifier input to the n hashing units ($32_1$, $32_2$, ... $32_n$) and the packet identifier output from said memory, and when the compared packet identifiers match, the forwarding information for the packet is obtained from said memory.

2. A lookup device (30) according to Claim 1, **characterized in** that each hashing unit ($32_1$, $32_2$, ... $32_n$) comprises a hash function means ($34_1$, $34_2$, ... $34_n$) for computing said first index, and a hash memory ($36_1$, $36_2$, ... $36_n$) connected to said hash function means ($34_1$, $34_2$, ... $34_n$).

3. A lookup device (30) according to Claim 2, **characterized in** that the lookup device (30) makes use of n different hash functions, one hash function for each hash functions means ($34_1$, $34_2$, ... $34_n$).

4. A lookup device (30) according to any one of Claims 1-3, **characterized in** that the n hashing units ($32_1$, $32_2$,

... $32_n$) are ordered by priority, wherein the first hashing unit ($32_1$) has the highest priority, and the n:th hashing unit ($32_n$) has the lowest priority.

5. A lookup device (30) according to Claim 4, **characterized in** that the first hash memory ($36_1$), representing the highest level in the lookup device (30), has the largest memory size, and the n:th hash memory ($36_n$), representing the lowest level in the lookup device, has the smallest memory size.

6. A lookup device (30) according to Claim 5, **characterized in** that the memory sizes for the n hash memories ($36_1$, $36_2$, ... $36_n$) are decreasing substantially lineary.

7. A lookup device (30) according to any one of Claims 1 - 6, **characterized in** that all the memories ($36_1$, $36_2$, ... $36_n$, 40; $36_1$, $36_2$, ... $36_n$) are Static Random Access Memories (SRAMs) and/or Dynamic Random Access Memories (DRAMs).

8. A lookup device (30) according to any one of Claims 2 - 7, **characterized in** that said first index function as an input to said hash memory ($36_1$, $36_2$, ... $36_n$) giving a packet identifier and forwarding information for the destination and a hit flag as outputs, and in that said lookup device (30) also comprises a selecting means (38) connected to the hit flag outputs of all n hash memories ($36_1$, $36_2$, ... $36_n$), a multiplexer (39) connected to the packet identifier and forwarding information outputs of all n hash memories ($36_1$, $36_2$, ... $36_n$), wherein said comparator (42) is connected to said multiplexer (39), wherein a set hit flag indicates that there was an entry in the hash memory ($36_1$, $36_2$, ... $36_n$) with the first index obtained by hashing the packet identifier, and the packet identifier from the hash memory ($36_1$, $36_2$, ... $36_n$) with the highest priority with the hit flag set, if any, is

used as input to said comparator (42), via said multiplexer (39), whereby said comparator (42) compares the packet identifier input to said hash function means ($34_1$, $34_2$, ... $34_n$) and the packet identifier output from said multiplexer (39), and when the compared packet identifiers match, the forwarding information for the packet is obtained from the hash memory ($36_1$, $36_2$, ... $36_n$) with the highest priority with the hit flag set.

9. A lookup device (30) according to any one of Claims 2 - 7, **characterized in** that said first index function as an input to said hash memory ($36_1$, $36_2$, ... $36_n$) giving a second index and a hit flag as outputs, and in that said lookup device (30) also comprises a selecting means (38) connected to the hit flag outputs of all n hash memories ($36_1$, $36_2$, ... $36_n$), a multiplexer (39) connected to the second index outputs of all n hash memories ($36_1$, $36_2$, ... $36_n$), an address memory (40), storing all packet identifiers tohgether with the forwarding information for the destination, connected to said multiplexer (39), wherein said comparator (42) is connected to said address memory (40), wherein a set hit flag indicates that there was an entry in the hash memory ($36_1$, $36_2$, ... $36_n$) with the first index obtained when hashing the packet identifier, and the second index from the hash memory ($36_1$, $36_2$, ... $36_n$) with the highest priority with the hit flag set, if any, is used as input to said address memory (40) giving a packet identifier and the forwarding information as outputs, whereby said comparator (42) compares the packet identifier input to the said hash function means ($34_1$, $34_2$, ... $34_n$) and the packet identifier output from said address memory (40), and when the compared packet identifiers match, the forwarding information for the packet is obtained from said address memory (40).

10. A method for classification and forwarding of packets, wherein each packet comprises a packet header

comprising a number of fields, wherein several fields in the packet header together form a packet identifier, wherein the method is **characterized by** the following steps:

- to compute, for each packet, a first index by hashing the input packet identifier in n different, parallel paths, wherein n is an integer and n≥2;
- and in dependence of the first index either directly or indirectly obtaining a packet identifier and forwarding information for the destination for said packet from one of at least n memories;
- to compare the input packet identifier and the packet identifier output from the memory; and
- if the compared packet identifiers match to make use of the forwarding information obtained from said memory.

11. A method according to Claim 10, **characterized in** that the computing step comprises the steps:

- to compute the first index by using n different hash functions, one hash function for each path; and
- to use the first index as an input to a table, one of n different tables.

12. A method according to any one of Claims 10 - 11, **characterized in** that the n paths are ordered by priority, wherein the first path has the highest priority, and the n:th path has the lowest priority.

13. A method according to Claim 12, **characterized in** that the first table, representing the highest level, has the largest size , and the n:th table, representing the lowest level, has the smallest size.

14. A method according to Claim 13, **characterized in** that the sizes of the n tables are decreasing substantially lineary.

15. A method according to any one of Claims 10 - 14, **characterized in** that each table stores packet identifiers and forwarding information for the destination, and wherein each table outputs a hit flag, wherein a set hit flag indicates that there was an entry in a table with the first index obtained by hashing the packet identifier, and the packet identifier from the table with the highest priority with the hit flag set, if any, is used as input to said comparing step.

16. A method according to any one of Claims 10 - 14, **characterized in** that said first index function as an input to said table giving a second index and a hit flag as outputs, wherein a set hit flag indicates that there was an entry in a table with the first index obtained when hashing the packet identifier, and the second index from the table with the highest priority with the hit flag set, if any, is used as input to an address memory giving a packet identifier as output, and said packet identifier is used as input to said comparing step.

17. A method according to any one of Claims 10 - 16, **characterized in** that, if a new packet identifier is to be added, it is initially hashed into the first path, and if a collision occurs, i.e. there is already a packet identifier with that first index in the first table, the two colliding packet identifiers are hashed into the second path, and if a collision occurs in the i:th path, the colliding packet identifiers are hashed into the (i+1):th path, wherein $1 \le i \le n-1$.

18. A method according to any one of Claims 10 - 17, **characterized in** that, if the compared packet identifiers not match, the method terminates for said packet identifier.

19. A method according to any one of Claims 15 - 16, **characterized in** that, if none of the n tables outputs a

set hit flag, the method terminates for said packet identifier.

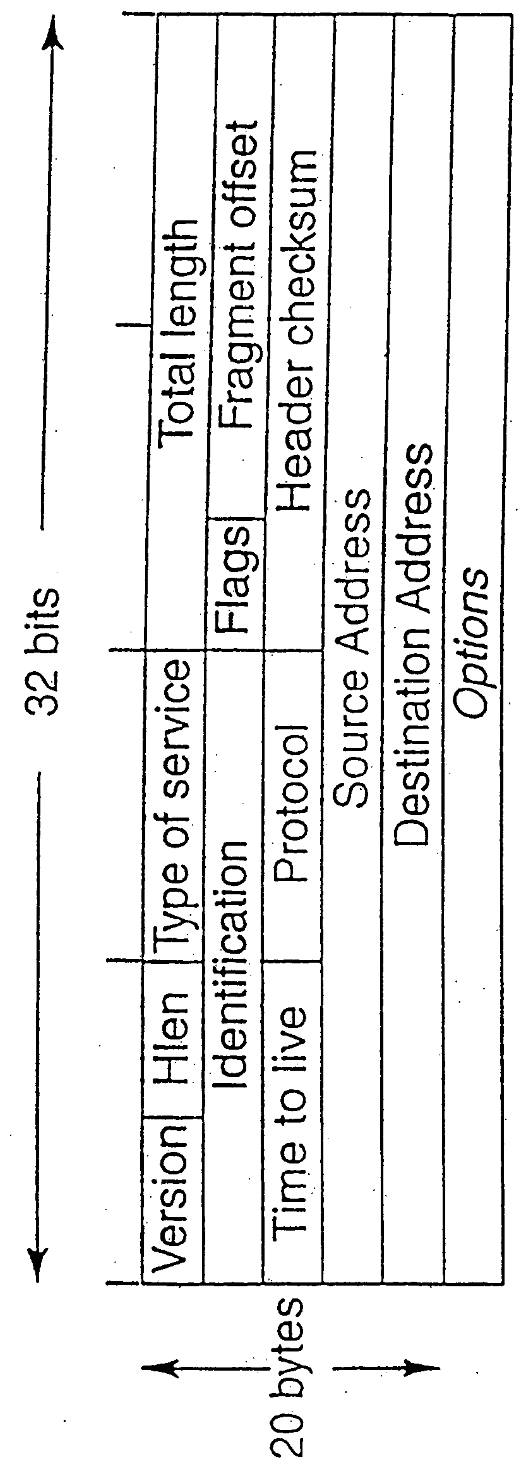
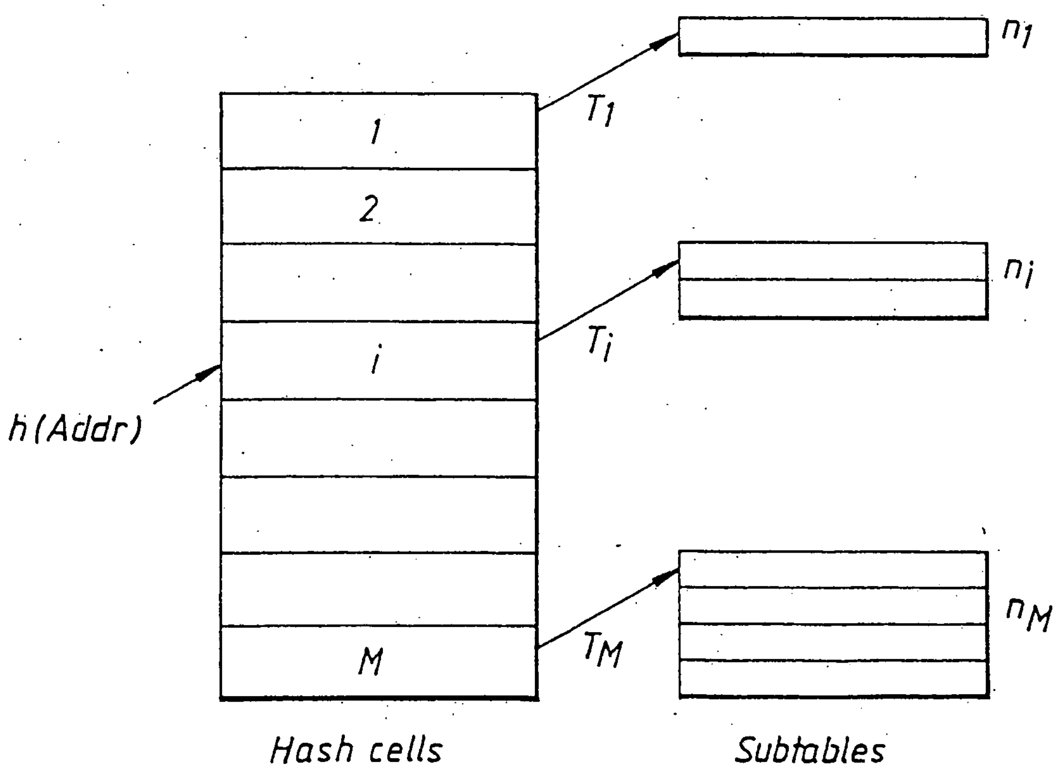

32 bits
Version
Hlen
Type of service
Total length
Identification
Flags
Fragment offset
Time to live
Protocol
Header checksum
Source Address
Destination Address
Options
20 bytes

Fig. 1

Fig. 2



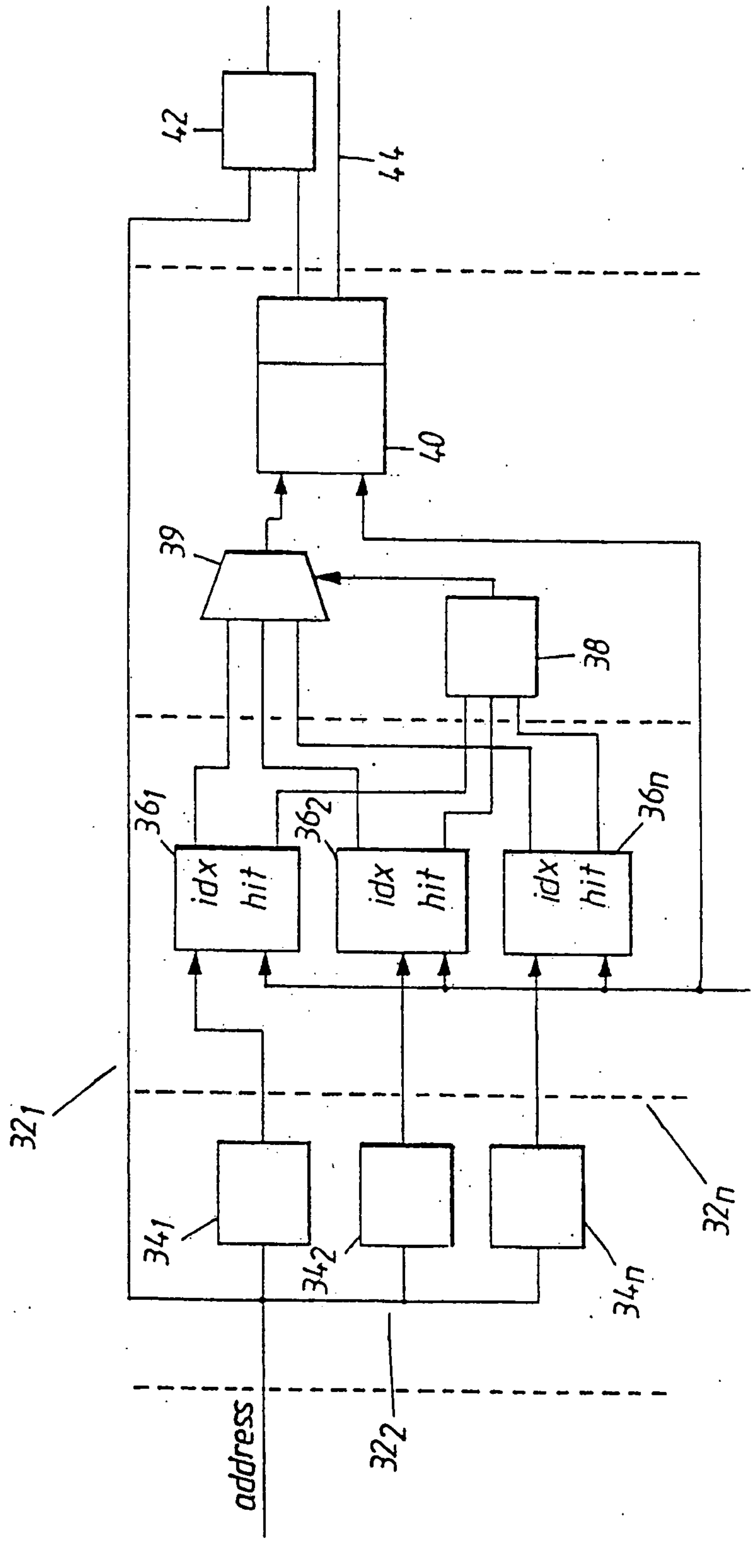
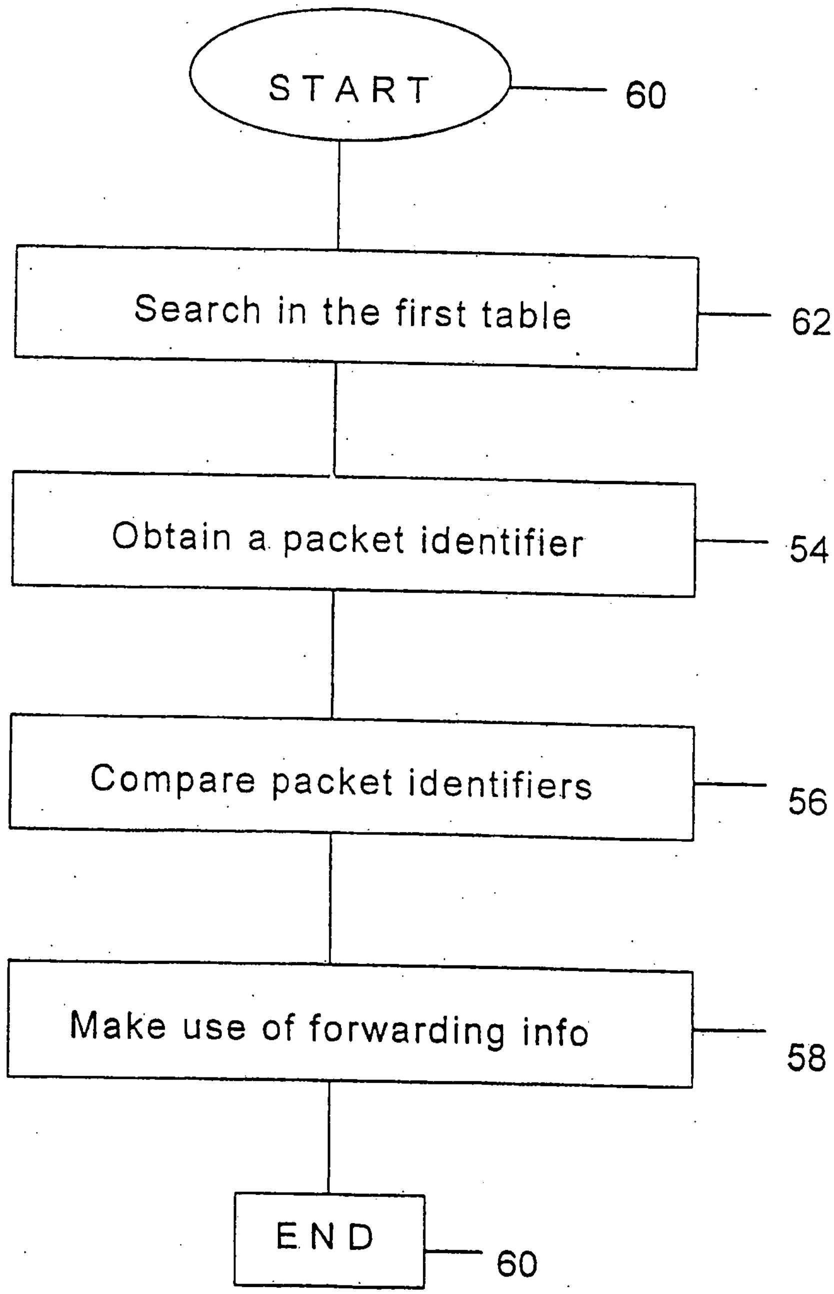

42
44
40
39
38
36₁
36₂
36n
idx
hit
idx
hit
idx
hit
32₁
34₁
34₂
34n
32n
322
address

Fig-2

Fig. 4


START
60
Search in the first table
62
Obtain a packet identifier
54
Compare packet identifiers
56
Make use of forwarding info
58
END
60

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04L 12/56

A3

(11) International Publication Number: WO 99/13620

(43) International Publication Date: 18 March 1999 (18.03.99)

(21) International Application Number: PCT/SE98/01585

(22) International Filing Date: 7 September 1998 (07.09.98)

(30) Priority Data:
9703293-2 9 September 1997 (09.09.97) SE

(71) Applicant (*for all designated States except US*): SICS [SE/SE]; Swedish Institute of Computer Science, Box 1263, S-164 29 Kista (SE).

(72) Inventors; and
(75) Inventors/Applicants (*for US only*): SJÖDIN, Peter [SE/SE]; Nyodlingsvägen 14B, S-191 40 Sollentuna (SE). MOESTEDT, Andreas [SE/SE]; Fatburskvarnsgata 2, S-118 64 Stockholm (SE).

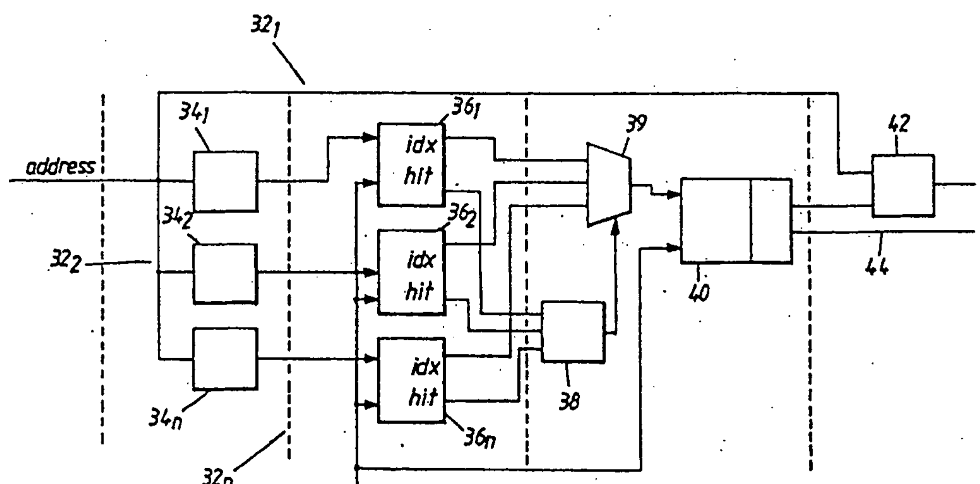
(74) Agent: ASKERBERG, Fredrik; L.A. Groth & Co. KB, P.O. Box 6107, S-102 32 Stockholm (SE).

(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, HR, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

**Published**
*With international search report.*
*Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.*

(88) Date of publication of the international search report: 3 June 1999 (03.06.99)

(54) Title: A LOOKUP DEVICE AND A METHOD FOR CLASSIFICATION AND FORWARDING OF PACKETS IN PACKET-SWITCHED NETWORKS




(57) Abstract

The present invention relates to a lookup device and a method for classification and forwarding of packets in packet-switched networks, wherein each packet comprises a packet header comprising a number of fields, wherein several fields in the packet header together form a packet identifier. The lookup device (30) comprises n parallel hashing units ($32_1$, $32_2$, ... $32_n$), wherein n is an integer and $n \geq 2$, for computing, for each packet, a first index by hashing the packet identifier, and in dependence of the first index either directly or indirectly obtaining a packet identifier and forwarding information for the destination for said packet from one of at least n memories . The n hashing units ($32_1$, $32_2$, ... $32_n$) are processing the same packet identifier at a time. The lookup device (30) also comprises a comparator (42) connected either directly or indirectly to at least one of said memories and to an input to said n hashing units ($32_1$, $32_2$, ... $32_n$) for comparing the packet identifier input to the n hashing ($32_1$, $32_2$, ... $32_n$) and the packet identifier output from said memory. When the compared packet identifiers match, the forwarding information for the packet is obtained from said memory.

*FOR THE PURPOSES OF INFORMATION ONLY*

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

| | |
|---|---|
| AL | Albania |
| AM | Armenia |
| AT | Austria |
| AU | Australia |
| AZ | Azerbaijan |
| BA | Bosnia and Herzegovina |
| BB | Barbados |
| BE | Belgium |
| BF | Burkina Faso |
| BG | Bulgaria |
| BJ | Benin |
| BR | Brazil |
| BY | Belarus |
| CA | Canada |
| CF | Central African Republic |
| CG | Congo |
| CH | Switzerland |
| CI | Côte d'Ivoire |
| CM | Cameroon |
| CN | China |
| CU | Cuba |
| CZ | Czech Republic |
| DE | Germany |
| DK | Denmark |
| EE | Estonia |
| ES | Spain |
| FI | Finland |
| FR | France |
| GA | Gabon |
| GB | United Kingdom |
| GE | Georgia |
| GH | Ghana |
| GN | Guinea |
| GR | Greece |
| HU | Hungary |
| IE | Ireland |
| IL | Israel |
| IS | Iceland |
| IT | Italy |
| JP | Japan |
| KE | Kenya |
| KG | Kyrgyzstan |
| KP | Democratic People's Republic of Korea |
| KR | Republic of Korea |
| KZ | Kazakstan |
| LC | Saint Lucia |
| LI | Liechtenstein |
| LK | Sri Lanka |
| LR | Liberia |
| LS | Lesotho |
| LT | Lithuania |
| LU | Luxembourg |
| LV | Latvia |
| MC | Monaco |
| MD | Republic of Moldova |
| MG | Madagascar |
| MK | The former Yugoslav Republic of Macedonia |
| ML | Mali |
| MN | Mongolia |
| MR | Mauritania |
| MW | Malawi |
| MX | Mexico |
| NE | Niger |
| NL | Netherlands |
| NO | Norway |
| NZ | New Zealand |
| PL | Poland |
| PT | Portugal |
| RO | Romania |
| RU | Russian Federation |
| SD | Sudan |
| SE | Sweden |
| SG | Singapore |
| SI | Slovenia |
| SK | Slovakia |
| SN | Senegal |
| SZ | Swaziland |
| TD | Chad |
| TG | Togo |
| TJ | Tajikistan |
| TM | Turkmenistan |
| TR | Turkey |
| TT | Trinidad and Tobago |
| UA | Ukraine |
| UG | Uganda |
| US | United States of America |
| UZ | Uzbekistan |
| VN | Viet Nam |
| YU | Yugoslavia |
| ZW | Zimbabwe |

# INTERNATIONAL SEARCH REPORT

International application No.
PCT/SE 98/01585

**A. CLASSIFICATION OF SUBJECT MATTER**

IPC6: H04L 12/56
According to International Patent Classification (IPC) or to both national classification and IPC

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)

IPC6: H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

SE,DK,FI,NO classes as above

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| A | US 5206856 A (FAN R.K. CHUNG), 27 April 1993 (27.04.93), column 2, line 36 - line 62, abstract<br>-- | 1-19 |
| A | EP 0563572 A2 (MOTOROLA, INC.), 6 October 1993 (06.10.93), column 2, line 1 - line 30, abstract<br>-- | 1-19 |
| A | IEEE/ACM TRANSACTIONS ON NETWORKING, Volume 4, No 2, April 1996, Girish P. Chandranmenon et al, "Trading Packet Headers for Packet Processing" page 141 - page 152<br>--<br>-------- | 1-19 |

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:
- "A" document defining the general state of the art which is not considered to be of particular relevance
- "E" erlier document but published on or after the international filing date
- "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- "O" document referring to an oral disclosure, use, exhibition or other means
- "P" document published prior to the international filing date but later than the priority date claimed
- "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- "X" document of particular relevance: the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- "Y" document of particular relevance: the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
- "&" document member of the same patent family

| Date of the actual completion of the international search | Date of mailing of the international search report |
|---|---|
| 8 April 1999 | 13 -04- 1999 |

| Name and mailing address of the ISA/ | Authorized officer |
|---|---|
| Swedish Patent Office<br>Box 5055, S-102 42 STOCKHOLM<br>Facsimile No. + 46 8 666 02 86 | Ewa Kowalska<br>Telephone No. + 46 8 782 25 00 |

# INTERNATIONAL SEARCH REPORT

Information on patent family members

02/03/99

International application No.
PCT/SE 98/01585

| Patent document cited in search report | Publication date | Patent family member(s) | Publication date |
|---|---|---|---|
| US 5206856 A | 27/04/93 | NONE | |
| EP 0563572 A2 | 06/10/93 | CA 2089823 A | 28/09/93 |
| | | JP 6104925 A | 15/04/94 |
| | | US 5365520 A | 15/11/94 |